

# Center-Shift: An Approach Towards Automatic Robust Mesh Segmentation (ARMS)

Mengtian Sun\*  
Purdue University

mengtian.sun@gmail.com

Yi Fang\*  
Purdue University

befangyi@gmail.com

Karthik Ramani  
Purdue University

ramani@purdue.edu

## Abstract

*In the area of 3D shape analysis, research in mesh segmentation has always been an important topic, as it is a fundamental low-level task which can be utilized in many applications including computer-aided design, computer animation, biomedical applications and many other fields. We define the automatic robust mesh segmentation (ARMS) method in this paper, which 1) is invariant to isometric transformation, 2) is insensitive to noise and deformation, 3) performs closely to human perception, 4) is efficient in computation, and 5) is minimally dependent on prior knowledge. In this work, we develop a new framework, namely the Center-Shift, which discovers meaningful segments of a 3D object by exploring the intrinsic geometric structure encoded in the biharmonic kernel. Our Center-Shift framework has three main steps: First, we construct a feature space where every vertex on the mesh surface is associated with the corresponding biharmonic kernel density function value. Second, we apply the Center-Shift algorithm for initial segmentation. Third, the initial segmentation result is refined through an efficient iterative process which leads to visually salient segmentation of the shape. The performance of this segmentation method is demonstrated through extensive experiments on various sets of 3D shapes and different types of noise and deformation. The experimental results of 3D shape segmentation have shown better performance of Center-Shift, compared to state-of-the-art segmentation methods.*

## 1. Introduction

### 1.1. Background

We have recently witnessed a booming growth in both creation and utilization of the 3D mesh data, which has attracted much interest across fields as diverse as engineering, biology, medicine, entertainment and so on [5, 6, 11,

12, 14, 16, 9]. Fostered by the rapid advancement in 3D data acquisition, conversion and visualization technologies, the utilization of 3D mesh data is now easier than ever before. With limited or no domain knowledge, nowadays people can easily create, search, browse and share 3D shape data, as evidenced by the rapid growth of the Google 3D warehouse. The ever-increasing amount of 3D mesh data poses an even greater challenge for the researchers. Amid all the research interests in this area, a fundamental low-level task is the segmentation. The segmentation technique is important because of its applicability towards potential applications including basic manufacturing practice tools, reverse engineering tools and preprocessing for advanced geometrical applications such as innovative 3D model generation [4, 21].

The segmentation problem of 3D mesh is challenging as the method is likely to encounter difficulties including: articulation, incompleteness, noise and deformations. Moreover, the segmentation should be as close to human perception as possible. It is also preferable that the method makes minimum assumptions about the object such as number of segments. In order to meet the above mentioned requirements, it is desirable for a segmentation method to be: 1) isometric transformation invariant, 2) noise and deformation insensitive, 3) computationally efficient, 4) result is perceptually consistent, and 5) demanding minimum user input. We define *Automatic Robust Mesh Segmentation* (ARMS) method as the segmentation method that meets all these requirements such that it is the overarching goal of all segmentation methods. Compared to existing methods, ARMS is much more reliable for producing meaningful segments and more informative. Hence ARMS can enable diverse geometric applications including shape retrieval, feature detection, skeleton extraction and mesh refinement.

### 1.2. Related work

Driven by continuing interest in this area, many methods have been proposed for 3D mesh segmentation [5, 11, 6, 12, 16]. While existing methods have shown potential for

---

\*Mengtian Sun and Yi Fang contributed equally to this paper.

approximating ARMS, some challenges still have not been fully addressed. While a complete review of the literature is beyond the scope of this article, we give a brief overview to focus on closely related works. Interested readers are referred to recent surveys [1, 18, 5] for an extensive overview of existing mesh segmentation methods.

The task of developing an ARMS method is non-trivial. Methods that depend on local features are not robust against noise due to not being aware of the global properties. In [11], the authors develop an articulation invariant segmentation method based on a hierarchial presentation of the body structure through core extraction. However, the segmentation is prone to instability as the feature points become unreliable under conditions of noise or deformation. The authors in [19] propose a segmentation method utilizing volume in the neighborhood of a single vertex in order to give a hierarchial segmentation, by solving the energy function minimization problem. This method is also sensitive to perturbation of local properties.

Recent research initiatives have lead to the discovery of methods that are more close to ARMS. The algorithms based on diffusion metrics [16, 6, 12, 15, 8] are promising as they explore the intrinsic structure of the 3D mesh by exploiting the rich yet implicit information encoded by diffusion kernel. The authors in [16] develop a deformation invariant shape signature, namely the *GPS*, based on eigenvalues and eigenfunctions of the Laplace-Beltrami operator (denoted as the Laplacian henceforth). Since segmentation is not the main focus of that work, it is achieved by simply applying K-Means on the *GPS* signature. In [6], the authors propose a hierarchial segmentation algorithm using diffusion maps, which produces consistent segments for articulated models. The authors in [6] adapts the diffusion maps to achieve hierarchical segmentation of articulated shapes. Persistence-based segmentation is recently introduced in [7]. While these methods can approximate ARMS to some extent, some difficulties still remain as evident in various aspects including the segment number determination, initial seeding and termination criteria.

### 1.3. Center-Shift for ARMS

We present the Center-Shift framework as our approach towards ARMS. We introduce a novel function defined on the mesh surface, the biharmonic kernel density function (*BDF*), which captures the intrinsic geometric features of the shape to a large extent. We define segment-exemplar as a representative vertex that is associated with a mesh segment, as represented by the orange colored nodes shown in Figure 1C and D. The Center-Shift algorithm is devised to seek stable segment-exemplars on the mesh surface by hunting maxima of the *BDF* feature space, in an automatic and robust manner. We show the pipeline of our method in Figure 1:

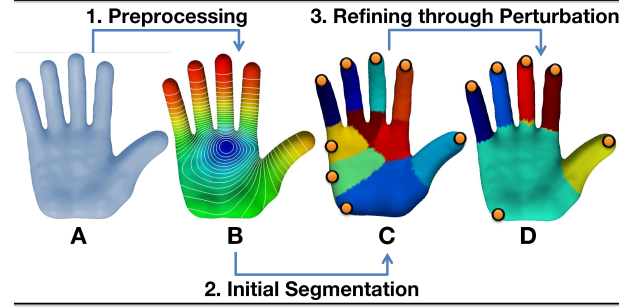


Figure 1. Center-shift segmentation framework pipeline. Nodes represent the segment-exemplars found during the Center-Shift process (only the visible ones are shown).

1. **Preprocessing** Given 3D mesh data, *BDF* is computed and mapped onto the mesh surface.
2. **Initial Segmentation** Center-shift is applied for preliminary segment-exemplar detection and mesh segmentation.
3. **Refining through perturbation** The candidate segment-exemplars detected in the previous step are merged through an effective perturbing-merging iterative process. The segmentation result is updated concurrently.

The contributions of this work towards Automatic Robust Mesh Segmentation (ARMS) method are: 1. We introduce biharmonic kernel density function (*BDF*) for characterizing the 3D shape. 2. A new approach (Center-Shift) for the determination of segment number, detection of segment-exemplars and formation of segments, all at the same time. 3. Extensive experiments on various datasets are carried out for the validation of the proposed framework of 3D mesh segmentation.

## 2. Method

### 2.1. Biharmonic kernel and biharmonic distance

In this section, we give a brief introduction to the biharmonic kernel and distance. Interested readers are referred to [13, 17] for more details. We represent the 3D shape as a graph  $G = (V, E, W)$ , where  $V$  is the set of all vertices,  $E$  is the set of all edges and  $W$  is the set of edge weights (cotangent weight is used). Biharmonic kernel is the Green's function of the biharmonic differential equation, which can be formulated as:

$$B(x, y) = \sum_{i>0} \frac{\phi_i(x)\phi_i(y)}{\lambda_i^2}, \quad (1)$$

Where  $B(x, y)$  denotes the biharmonic kernel value between points  $x$  and  $y$ ,  $\lambda_i$  denotes the  $i$ th eigenvalue and  $\phi_i$

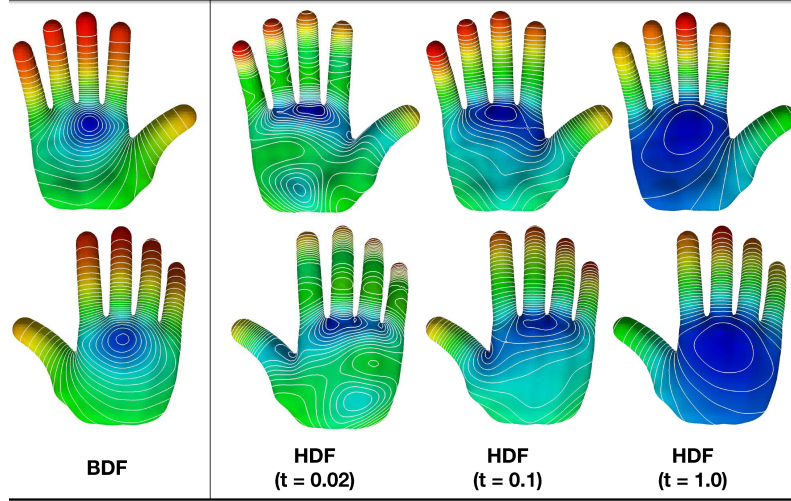


Figure 2. Comparison of biharmonic kernel density function (*BDF*) and heat kernel density function (*HDF*) computed at different diffusion times ( $t = 0.02, 0.1$  and  $1.0$ ). Both front and back of the 3D hand model are color mapped according to the respective function value distribution. Red color indicates high function value, blue color indicates low function value and other colors represent values in between (logarithmic scaled). The isolines follow the distributions of function values.

denotes the  $i$ th eigenvector of the Laplacian. Consequently, the biharmonic distance is introduced as a robust and accurate distance measure on the mesh surface, which can be formulated as:

$$D(x, y)^2 = g(x, x) + g(y, y) - 2g(x, y) \quad (2)$$

where  $D(x, y)$  denotes the biharmonic distance between points  $x$  and  $y$ . The discrete construction of the biharmonic kernel and distance are introduced in [13, 17]. Note that, biharmonic kernel in this article refers to the global biharmonic kernel in the original work [17].

The advantage of the biharmonic kernel arises from the the denominator  $\lambda^2$ , whereas the heat kernel has  $e^{-\lambda_i t}$ . The denominator balances the decaying rate of the normalized eigenvalue ( $1/\lambda_k^2$  in this case). This allows the biharmonic kernel to exploit local information and explore global properties of the 3D mesh surface at the same time. The biharmonic kernel is intrinsic, informative, robust and computationally efficient [17].

## 2.2. Biharmonic kernel density function

We define biharmonic kernel density function (*BDF*) on the 3D mesh domain  $\mathcal{M}$ , which is formulated as:

$$BDF(x) = \{B(x, x) | x \in \mathcal{M}\} \quad (3)$$

where  $B$  is the biharmonic kernel matrix.

*BDF* provides a highly concise representation of the 3D mesh without compromising loss of critical geometrical attributes. The validity of *BDF* is supported by the Informative Theorem proposed by Sun et. al. in [20]. According to the theorem, the diagonal of the kernel matrix is almost as

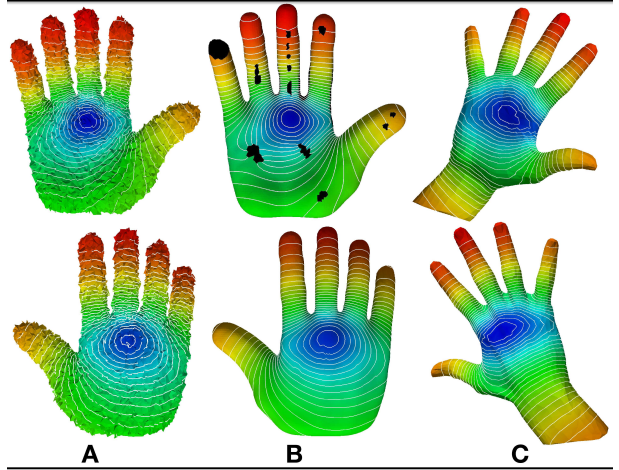


Figure 3. Robustness test of *BDF*. Figure A and B demonstrate the hand model used in Figure 2 with numerical noise and holes, respectively. Figure C shows another hand model with very different topological structure. *BDF* is insensitive to noise and deformations, and is comparable on models with similar shape.

informative as the original kernel. The theorem holds under the condition that the Laplacian has non-repeated eigenvalues, which is true throughout our test cases. As a result, *BDF* naturally inherits many of the properties of the biharmonic kernel, which makes it very lucrative for geometric applications. We demonstrate the properties of *BDF* through two experiments:

**Informative** For the purpose of comparison, we define heat kernel density function (*HDF*) as the counterpart of *BDF* in the widely used heat kernel. In other words, *HDF* val-

ues correspond to the diagonal elements of heat kernel. We compare  $BDF$  with  $HDF$  calculated at different diffusion times as shown in Figure 2. It is observed that at small time ( $t = 0.02$ ),  $HDF$  is localized without global awareness, evidenced by the presence of many local minima. At a larger time ( $t = 0.1$ ),  $HDF$  isolines become more shape-aware as seen on the fingers, however still remain largely localized as seen on the center of the hand. The  $HDF$  isolines appear to be similar to that of  $BDF$  after heat has explored the entire surface for a large time ( $t = 1.0$ ). However,  $BDF$  is more globally and locally aware than  $HDF$ , as can be observed from the center of the hand:  $BDF$  isolines are denser than that of  $HDF$  which indicates a better parametrization of the shape. Also,  $BDF$  isolines are near circles while  $HDF$  produces elliptical isolines.

**Robust** We regenerate the hand model used in Figure 2 with numerical noises and holes, as shown in Figure 3A and 3B. The distribution of  $BDF$  values of the new models remain largely unchanged comparing to the result in Figure 2. The isolines follow the shape indifferently under the condition of noise and holes.

**Consistent** We show  $BDF$  of another hand model with very different sampling and topological structure in Figure 3C. In spite of the differences in shape, the geometry features captured by  $BDF$  of the two hand models are highly consistent. Note that  $BDF$  value distribution of the new hand model has an extra mode because the kernel intelligently senses the extra wrist part. Other than that, the geometry features captured by  $BDF$  remain largely unaffected throughout the rest of the mesh surface.

## 2.3. Center-shift Framework

### 2.3.1 BDF feature space

We define the  $BDF$  feature space where every vertex on the mesh is associated with the corresponding  $BDF$  value. As observed from the hand model in Figure 2, the distribution of  $BDF$  values on the mesh surface reveals the intrinsic structure of the model: The maxima are located at geometrical salient points such as finger tips. The isolines lie perpendicularly to the axis of individual part and follow the shape intelligently. We learn from these observations that the modes of  $BDF$  feature space can be suitably used as segment-exemplars. Consequently, the clusters of vertices can be delineated based on the structure of this multimode distribution. The problem of mesh segmentation is naturally converted to the clustering problem of assigning vertices to the associated segment-exemplars, which correspond to the modes of  $BDF$  feature space.

### 2.3.2 Center-Shift Algorithm

We hereby propose the Center-Shift algorithm for the detection of segment-exemplars by searching the modes of  $BDF$

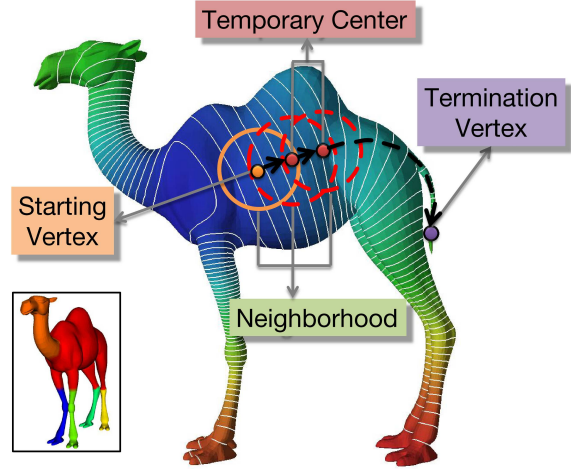


Figure 4. Center-shift applied on single vertex. The center figure illustrates the process of Center-Shift of single vertex on a camel mesh model (color mapped with  $BDF$ ). The orange colored node and ring represent the starting vertex and its neighborhood. The red colored nodes and rings represent the temporary centers and their corresponding neighborhoods. The above depicted Center-Shift process terminates after arriving at the termination vertex (purple node on the tail). The lower left image shows the segmentation result which contains 6 segments.

feature space, which describes the characteristic of vertices on the mesh surface, analogous to Mean-Shift that explores the gaussian feature space. Let  $G_i$  denotes the neighborhood of the  $i$ th vertex  $V_i$ .  $G_i$  is the set of all vertices that are within a specified distance  $r$  from  $V_i$ , where  $r$  is denoted as the neighborhood radius. The common practice of using a universal parameter value for  $r$  is suboptimal as individual vertex's property is overlooked. On the contrary, we take an adaptive approach towards neighborhood construction by considering the distance distribution from all vertices to the given vertex. Our neighborhood radius computation is given by:

$$r_i = h \cdot \max\{D(V_i, V_j) | j \in \mathcal{M}\} \quad (4)$$

where  $h$  is denoted as the neighborhood threshold and  $D(V_i, V_j)$  denotes the biharmonic distance between vertices  $V_i$  and  $V_j$ .  $G_i$  is then constructed based on  $r_i$ .

We visualize the process of applying Center-Shift algorithm on a vertex as in Figure 4. Starting from a vertex, we calculate the weighted mean of its neighborhood using coordinates weighted by  $BDF$  values. Due to the nature of this computation, one critical issue is that the weighted mean may not be precisely on the mesh surface. We address this problem by replacing the weighted mean with the closest vertex on the mesh surface (denoted as the temporary center). If the temporary center is the same as the starting vertex, the process terminates as the starting vertex is already the barycenter of its own neighborhood weighted by



$BDF$ . Otherwise, the above process is repeated by starting from the temporary center. The vertex at which the iteration ends is referred to the termination vertex and considered as a potential segment-exemplar. The Center-Shift algorithm approaches the mode of feature space, which is associated with a potential segment-exemplar, by iteratively shifting to regions of higher  $BDF$ . The starting vertex is assigned to the segment associated with the segment-exemplar detected. The numerical implementation is described by Algorithm 1.

---

**Algorithm 1** CENTER-SHIFT (*starting-vertex*)

---

- 1:  $\triangleright$  Given vertices number  $N$ , coordinates *coord* and bi-harmonic distance matrix  $D$
  - 2:  $\triangleright G_{sv}$  represents the neighborhood of *starting-vertex*
  - 3: *weighted-mean*  

$$= \sum_{i \in G_{sv}} coord(i)BDF(i) / \sum_{i \in G_{sv}} BDF(i)$$
  - 4:  $\triangleright$  find the closest vertex to *weighted-mean*
  - 5: *temporary-center*  

$$= \arg \min_{i \in \mathcal{M}} \|weighted-mean - coord(i)\|$$
  - 6: **while**  $D(temporary-center, starting-vertex) > converge-threshold$   
**and** not reach maximum iteration number **do**
  - 7:   CENTER-SHIFT(*temporary-center*)
  - 8: **end while**
  - 9: *termination-vertex* = *temporary-center*
  - 10: *exemplar* = *termination-vertex*
  - 11:  $\triangleright$  return the termination vertex as a candidate segment-exemplar
  - 12: return *exemplar*
- 

### 2.3.3 Center-Shift driven mesh segmentation

We propose the following two-step mesh segmentation method based on Center-Shift algorithm:

**1. Initial segmentation** We start with applying Center-Shift algorithm on every vertex on the mesh. Each termination vertex becomes a candidate segment-exemplar with the corresponding starting vertex being a member of its associated segment. In this way, vertices are assigned to the same cluster if they share the same termination vertex. However, the above described segmentation method usually leads to over-segmentation (as in Figure 1C) because the shifting process is likely to be trapped by local maxima, especially under condition of noise and deformation.

**2. Refining through perturbation** We overcome over-segmentation by seeking stable segment-exemplars within the candidates obtained in the previous step. To examine if a candidate segment-exemplar is stable, we simply perturb the candidate within its neighborhood and apply Center-Shift on the perturbed vertex. The candidate segment-

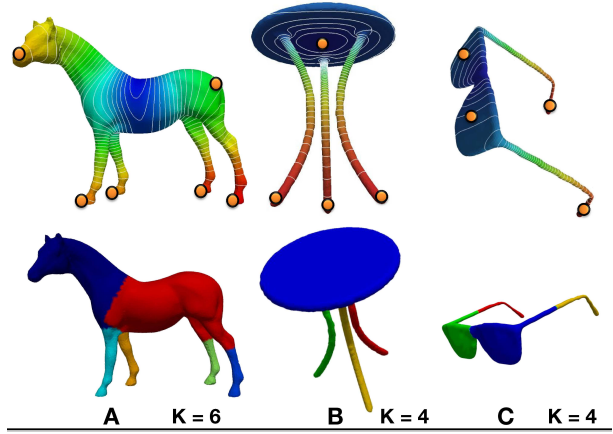


Figure 5. Segment-exemplar detection and segmentation experiment result. Figures in the first row illustrate the shapes color mapped with  $BDF$  along with the detected stable segment-exemplars (orange colored nodes). The segmentation results are shown in the second row.  $K$  denotes both the number of stable segment-exemplars and segments.

exemplar is considered stable if it coincides with the new segment-exemplar returned by Center-Shift. Otherwise the candidate segment-exemplar is eliminated and its associated vertices are re-assigned to other segment-exemplars, along with which the number of segments is reduced and the segmentation result is refined. The above described process is iterated for every candidate segment-exemplar till convergence. Here convergence is defined as the state that the segment-exemplars remain unchanged for a consecutive number of iterations (denoted as stable iteration number). The remaining segment-exemplars are thus denoted as stable segment-exemplars. This iteration process is computationally efficient, as we only investigate the segment-exemplars, whose number is orders of magnitude smaller than  $N$  and is being continuously reduced. Empirically, we use stable iteration number 20. In addition, the refining process largely reduces our dependence on the selection of the neighborhood threshold  $h$  as the final set of stable segment-exemplars are insensitive to the choice of parameter value of  $h$ . We use a neighborhood threshold value around 0.1 throughout the tests. The implementation of the refinement step is presented in Algorithm 2.

We show results of segment-exemplar detection and segmentation in Figure 5. As shown in the figure, our algorithm automatically determines the number of segments, detects the segment-exemplars and produces perceptually consistent segments across different types of 3D models.

## 3. Experiment Results

We demonstrate the performance of the Center-Shift segmentation method through extensive experiments on various shape datasets and over types including articulation,

---

**Algorithm 2** REFINEMENT

---

```

1: ▷ Given the candidate segment-exemplars exemplars,
   vertices number N and coordinates coord
2: while not converged and not reach maximum iterations
   do
3:   ▷ K is the number of elements in exemplars
4:   create array perturbed-vertices of length K
5:   for i = 1 to K do
6:     perturbed-vertices(i) is a randomly chosen ver-
       tex within  $G_i$  other than exemplars(i)
7:     new-exemplars(i)
       = CENTER-SHIFT(perturbed-vertices(i))
8:     if new-exemplars(i)  $\neq$  exemplars(i) then
9:       reassign vertices associated with exemplars(i)
         to the nearest new-exemplars
10:    end if
11:  end for
12:  update exemplars
13: end while
14: return exemplars

```

---



Figure 6. Segmentation results of articulated models. Group A and B consist of armadillo and centaur models with various poses, respectively. More examples of armadillo model with pose variation can be found in Figure 7 and 8.

incompletion, noise, topological short-circuit and other deformations. The test data used in our experiments are from TOSCA dataset [2], SHREC dataset [3] and Stanford segmentation benchmark [5].

### 3.1. Consistency over articulation

In this experiment, we test the performance of Center-Shift on models with articulation changes. We carry out two sets of experiments on the armadillo models and the centaur models, assuming only isometric transformation among shapes within the same group. The segmentation results are demonstrated in Figure 6, where we observe two interesting results. First, within the same group, models have the same number of segment (6 for group A and 9

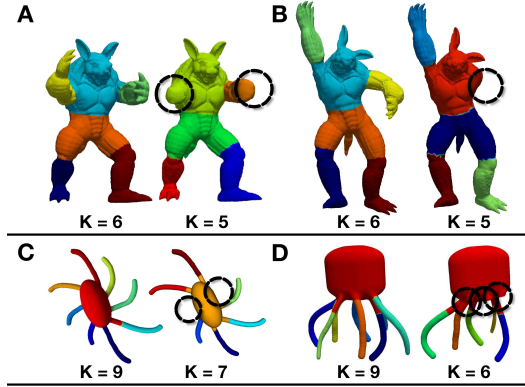


Figure 7. Segmentation results of incomplete models. Incomplete models are shown by the right side of its original version. Black colored circle denotes missing part. *K* denotes the number of segments.

Center-Shift	Shape Diameter	Core Extraction	Randomized Cuts
K = 6	K = 17	K = 8	K = 7
K = 6	K = 16	K = 9	K = 7

Figure 8. Comparison with other segmentation methods. In this figure, non-touching patches represent different segments regardless of the color.

for group B). The number of segments determined by our method is consistent throughout the same group, without prior knowledge of the group or individual shape. Second, Center-Shift correctly segments the meaningful functional components of the objects, such as tail and limbs. Moreover, the cuts between segments are consistent throughout the same group of models. More examples of armadillo models with pose variations can be found in Figure 7 and 8. The segmentation results obtained by our method are both consistent and visually salient for articulated shapes.

### 3.2. Consistent segmentation of incomplete shapes

We are also interested in the sensitivity of the segmentation method over incomplete shapes. Therefore, we apply Center-Shift on a set of complete and incomplete models as shown in Figure 7. For each group, the segmented incomplete model is displayed along with its original version. The incompleteness can be identified as body parts missing from the original model such as claws of the armadillo and arms of the octopus. The Center-Shift framework automatically

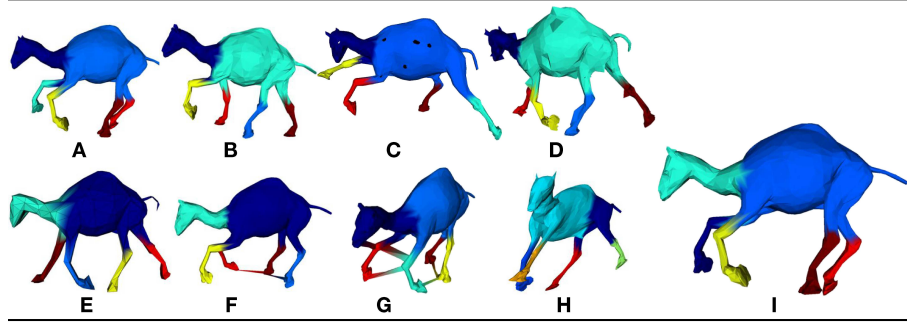


Figure 9. Noise and deformation model segmentation result. 8 different types of noisy and deformed models from the SHREC database are tested, following the terminology of the original work: A. null, B. noise, C. hole, D. shot noise, E. sampling, F. topology (2 linkages), G. topology(7 linkages), H. affine, I. scale.

determines the number of segments, without prior knowledge of the incomplete model or reference to the complete model. Notice that in Figure 7A, the algorithm correctly identifies the incomplete left claw even a large portion of it is missing, whereas the right claw is almost completely absent and is thus not identified as a component. Figure 7C and D further demonstrate the performance of Center-Shift when multiple parts are missing from the original model. Our method correctly predicts the number of segments for the incomplete model regardless of the number of missing components. Moreover, the partitions of the rest of the incomplete model are almost unaffected. In this test, the same neighborhood threshold is used throughout each group. Through the incompleteness test, Center-Shift is demonstrated to be consistent, part-aware, and operates under minimum number of assumptions (the neighborhood threshold in this work).

### 3.3. Insensitive to noise and deformation

We demonstrate the robustness of Center-Shift under noise and deformation. The test dataset is a group of camel models from the SHREC database. Our test results are shown in Figure 9. In this figure, model A serves as a reference for the test dataset, which itself retains the same segmentation as the other camel model in Figure 4 regardless of their very different sampling and triangulation. Figure 9B, C and D demonstrate models with noise, holes and shot noise, where the segmentation is unaffected from observation. Figure 9E shows model with 'sampling' transformation, which consists of only 626 vertices (approximately 25% of model A). The fact that the segmentation of this highly simplified model is consistent with the reference model demonstrates the applicability of our algorithm on sparse meshes. It is also shown that our method is insensitive to topological changes as observed in Figure 9F and G, where the topological structure is largely changed due to the linkages. Model H has affine transformation. Under the condition of transformed distance measure due to

this severe non-isometric deformation, our method still correctly identifies geometrical features of the shape and delivers reasonable segmentation result. The segmentation also persists for the scaled model H, which is larger than model A. Throughout the robustness test, our segmentation results are consistent both in segment number and the segmented functional components. Our experimental results demonstrate that Center-Shift is insensitive to noisy and deformed models.

### 3.4. Comparison with other segmentation methods

We compare Center-Shift with other segmentation methods, including: Shape Diameter (SD) [19], Core Extraction (CE) [11] and Randomized Cuts (RC) [10]. Readers can find detailed review on these methods in [5]. As noted in the review, both SD and CE predict number of segments while RC is the best performer out of the 7 machine segmentation methods reviewed. We reconstruct the segmentation result of these methods from the Stanford Segmentation Benchmark data. As shown in Figure 8, segmentation results of our method is the most consistent and perceptually acceptable. The number of segments predicted by SD or CE are not consistent for models from the same group as they use local property of the mesh, which is unreliable. For RC, as it requires input of number of segments, we use the result that has the closest  $K$  as our estimation. As pointed out in the original article, RC tends to produce unstable results for symmetric models as it prefers one of multiple cuts that are of similar costs (claw and leg). RC also requires the longest computation time as reported in the survey [5].

## 4. Discussion and conclusion

In this work we develop a previously undescribed Center-Shift framework as our approach towards Automatic Robust Mesh Segmentation (ARMS). The Center-Shift explores the  $BDF$  feature space associated with the 3D mesh, and accomplishes the tasks of the segment number determination, segment-exemplar detection and mesh segmen-

tation, all at the same time and with minimum user input. The resulting segmentation is perceptually consistent and insensitive to noise and deformation. However, we note that Center-Shift should not be applied blindly. We used the neighborhood threshold to control the scope of neighborhood exploration. a larger threshold value enables more global exploration of the shape structure and a smaller threshold value leads to more exploitation of the local information. The Center-Shift framework is less dependent on user selected values than other methods as the only parameter is the neighborhood threshold. The sensitivity to this parameter is further diminished by the refining process through perturbation. In the future, we will further explore the optimal threshold value for exploring 3D shape in our computational framework. Based on empirical results, the Center-Shift prefers a threshold value that is not too small that can be trapped in local maxima or too large that results in under-segmentation. Our choice of threshold value performs reasonably well throughout the experiments. In the future, we will further explore the refinement of the cuts to make our segmentation even more perceptually consistent.

## Acknowledgement

We would like to thank the Institute for Pure & Applied Mathematics (IPAM), a National Science Foundation (NSF) Math Institute at the University of California at Los Angeles (UCLA) for the fellowship to support Karthik Ramani. This work is partly based upon work supported by the National Science Foundation Partnership for Innovation Grant# 0917959 (3D Hub), NSF IIS Grant# 0535256, Donald W. Feddersen Chair Professorship support, Purdue University support for Faculty Study in a second discipline, and the School of Mechanical Engineering. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Also we thank Professor Raif Rustamov for insightful comments on our paper.

## References

- [1] M. Attene, S. Katz, M. Mortara, G. Patane, M. Spagnuolo, and A. Tal. Mesh segmentation - a comparative study. *Shape Modeling and Applications*, 2006. SMI 2006. *IEEE International Conference on*, pages 7–7, 2006.
- [2] A. M. Bronstein, M. M. Bronstein, and R. Kimmel. Numerical geometry of non-rigid shapes. *Springer*, 2008.
- [3] A. M. Bronstein, M. M. Bronstein, R. Kimmel, M. Mahmoudi, and G. Sapiro. A gromov-hausdorff framework with diffusion geometry for topologically-robust non-rigid shape matching. *Int. J. Comput. Vision*, 89:266–286, 2010.
- [4] S. Chaudhuri, E. Kalogerakis, L. Guibas, and V. Koltun. Probabilistic reasoning for assembly-based 3d modeling. *ACM Trans. Graph.*, 30:35:1–35:10, August 2011.
- [5] X. Chen, A. Golovinskiy, and T. Funkhouser. A benchmark for 3D mesh segmentation. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 2009.
- [6] F. De Goes, S. Goldenstein, and L. Velho. A hierarchical segmentation of articulated bodies. *Computer Graphics Forum*, 27:1349–1356, 2008.
- [7] T. Dey, K. Li, C. Luo, P. Ranjan, I. Safa, and Y. Wang. Persistent heat signature for pose-oblivious matching of incomplete models. *Computer Graphics Forum*, 29:1545–1554, 2010.
- [8] Y. Fang, M. Sun, M. Kim, and K. Ramani. Heat-mapping: A robust approach toward perceptually consistent mesh segmentation. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, pages 2145–2152, 2011.
- [9] Y. Fang, M. Sun, and K. Ramani. Temperature distribution descriptor for robust 3d shape retrieval. *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*, pages 9–16, june 2011.
- [10] A. Golovinskiy and T. Funkhouser. Randomized cuts for 3d mesh analysis. *ACM Trans. Graph.*, 27:145:1–145:12, December 2008.
- [11] S. Katz, G. Leifman, and A. Tal. Mesh segmentation using feature point and core extraction. *The Visual Computer*, 21:649–658, 2005. 10.1007/s00371-005-0344-9.
- [12] Y.-K. Lai, S.-M. Hu, R. R. Martin, and P. L. Rosin. Fast mesh segmentation using random walks. *Proceedings of the 2008 ACM symposium on Solid and physical modeling*, pages 183–191, 2008.
- [13] Y. Lipman, R. M. Rustamov, and T. A. Funkhouser. Biharmonic distance. *ACM Trans. Graph.*, 29:27:1–27:11, July 2010.
- [14] R. Osada, T. Funkhouser, B. Chazelle, and D. Dokin. Shape distributions. *ACM Transactions on Graphics*, 33:133–154, 2002.
- [15] M. Reuter. Hierarchical shape segmentation and registration via topological features of laplace-beltrami eigenfunctions. *International Journal of Computer Vision*, 89:287–308, 2010. 10.1007/s11263-009-0278-1.
- [16] R. M. Rustamov. Laplace-beltrami eigenfunctions for deformation invariant shape representation. *Proceedings of the fifth Eurographics symposium on Geometry processing*, pages 225–233, 2007.
- [17] R. M. Rustamov. Multiscale biharmonic kernels. *Computer Graphics Forum*, 30(5):1521–1531, 2011.
- [18] A. Shamir. A survey on mesh segmentation techniques. *Computer Graphics Forum*, 27(6):1539–1556, 2008.
- [19] L. Shapira, A. Shamir, and D. Cohen-Or. Consistent mesh partitioning and skeletonisation using the shape diameter function. *The Visual Computer*, 24:249–259, 2008. 10.1007/s00371-007-0197-5.
- [20] J. Sun, M. Ovsjanikov, and L. Guibas. A concise and provably informative multi-scale signature based on heat diffusion. *Computer Graphics Forum*, 28(5):1383–1392, 2009.
- [21] H. Yamauchi, S. Lee, Y. Lee, Y. Ohtake, A. Belyaev, and H.-P. Seidel. Feature sensitive mesh segmentation with mean shift. *Shape Modeling and Applications, International Conference on*, 0:238–245, 2005.